

# eboxdetect

February 23, 2011

## Abstract

Sliding box EPIC source searching using both local or global background.

## 1 Instruments/Modes

Instrument	Mode
EPIC MOS:	IMAGING
EPIC PN:	IMAGING

## 2 Use

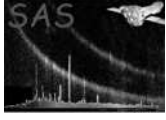
pipeline processing	yes
interactive analysis	yes

## 3 Description

The **eboxdetect** task has two different modes of operation: (1) local detection and (2) map detection. **eboxdetect** is first run in local mode. The local mode source lists serves as input for task **esplinemap** which creates background maps (one per energy band for each EPIC instrument) which are then used to run **eboxdetect** in map mode.

The boolean input parameter **usemap** controls whether the program operates in map mode or in local mode. In both modes, source searching is only performed in the area of the images which is marked by an optional detection mask created by task **emask**. Use of a detection mask is controlled by the boolean input parameter **withdetmask**. If true, one detection mask for each EPIC instrument must be supplied (parameter **detmasksets**). Parameter **withexpimage** controls the optional use of exposure maps as created by task **eexpmap**. The value of the exposure image at the location of each detected source is then used to calculate vignetting- and deadtime corrected source count rates. If parameter **withexpimage** is set to true one exposure image for each energy band of each EPIC instrument must be supplied (parameter **expimagesets**). If no exposure images are supplied raw count rates are calculated by deriving the exposure information from the good time intervals.

**eboxdetect** may be used to perform source detection on individual images or on sets of images from different energy bands and/or different EPIC instruments. If source detection on multiple images is



performed, the input images and corresponding exposure images, background images and detection masks have to be supplied as lists of file names. All input images and accompanying exposure images, background images and detection masks must have identical orientation and binning and must be supplied in a consistent order.

1. Local detection mode: The purpose of the local detection step is to provide an input list of source positions for task **esplinemap** which then constructs a background map from the non-source locations. Source counts are accumulated from a 3 x 3 or 5 x 5 (controlled by parameter **boxsize**) pixel window and the background is determined from the surrounding 40 ( 7 x 7 pixel window) or 56 pixels (9 x 9 pixel window), respectively. Detection of moderately extended objects (up to several times the PSF size) is achieved by searching the image in up to three consecutive detection runs each doubling the pixel size. The input image pixel size resampled from the original detector pixels is chosen such that the detection window corresponds to the size of the on-axis PSF. It is thus required that the PSF is at least moderately oversampled (which is the case for both pn and MOS CCD EPIC data). It is recommended to use a detection threshold of **likemin**=8 to provide a complete source list as input for **esplinemap**.
2. Map detection mode: Source counts are accumulated from a 3 x 3 or 5 x 5 pixels window with the option to use a position and energy dependent PSF weighted filter (matched filter; not yet implemented). Detection of extended objects is again achieved by doubling the pixel sizes in up to three consecutive detection runs. In map detection mode the background is taken from the background maps determined by **ESPLINEMAP**, resulting in an improved detection sensitivity as compared to the local detection step. If the map detection run is used as input for the for **emldetect**, it is recommended to use a somewhat lower detection threshold than for **emldetect**, e.g. **mlmin**=10 for **emldetect** and **likemin**=8 for **eboxdetect**.

Both in local mode and map mode background subtracted source counts are calculated by applying correction factors to account for the respective fractions of source counts falling in each source and background area. The respective off-axis angle dependent correction factors are calculated using the current calibration PSF (medium accuracy model). The following equations describe the PSF correction of source and background counts as implemented in the code of the task.

$n$  = detection box size

Enboxed energy fraction in source box:

$$\alpha = \sum_{n \times n} \text{PSF}$$

Fraction of source counts in background counting area:

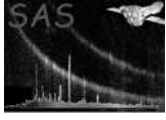
$$\beta = \sum_{(n+4) \times (n+4)} \text{PSF} - \sum_{n \times n} \text{PSF}$$

Raw box counts:

$$BOX\_CTS = \sum_{n \times n} \text{image}$$

Raw background map:

$$BG\_RAW = \left( \sum_{(n+4) \times (n+4)} \text{image} - \sum_{n \times n} \text{image} \right) / ((n+4)^2 - n^2)$$



In local detection mode:

PSF corrected and background subtracted source counts:

$$SCTS = \frac{BOX\_CTS - BG\_RAW \cdot n^2}{\alpha - \beta \cdot n^2 / ((n+4)^2 - n^2)}$$

PSF corrected background map:

$$BG\_MAP = BG\_RAW - SCTS \cdot \beta / ((n+4)^2 - n^2)$$

$$err_{bg} = \sqrt{BG\_RAW \cdot ((n+4)^2 - n^2) / ((n+4)^2 - n^2)}$$

Error of source counts:

$$SCTS\_ERR = \frac{\sqrt{BOX\_CTS + (n^2 \cdot err_{bg})^2}}{\alpha - \beta \cdot n^2 / (n+4)^2}$$

In map detection mode:

$$SCTS = (BOX\_CTS - BG\_MAP \cdot n^2) / \alpha$$

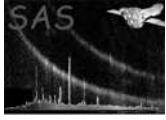
The resulting output source table contains one row per input image for each detected source, plus a number of summary rows containing the broad band results for each EPIC telescope and the combined results for all EPIC telescopes taken together. The individual source rows are identified through the column entries ID\_INST and ID\_BAND in the output table where ID\_INST refers to the EPIC instrument (1: PN, 2: MOS1, 3: MOS2, 0: summary row) and ID\_BAND is the energy band number as defined by the ordering in which the energy bands are given on the command line (an ID\_BAND value of 0 again refers to the summary information). No summary rows are output if source detection is only performed on a single input image.

The source table lists statistical errors for both count rates and source positions. Count rate errors are calculated by assuming Poissonian statistics ( $error = \sqrt{counts}$ ) in both the source and background cells (if eboxdetect is run in local mode) and by applying standard error propagation. If run in map mode the background taken from the spline background maps is assumed to be free of statistical errors. Positional errors are assumed to be equal to the standard deviation of the distribution of the counts in the detection cell. The errors of the derived parameters, such as count rates, fluxes, and source positions in celestial coordinates are derived from the count and image pixel positional errors, respectively.

Following the definition which was, e.g., used by the ROSAT mission, detection likelihoods (per energy band and total) are given for each source in the form  $L = -\ln p$  where  $p$  is the probability of Poissonian random fluctuation of the counts in the detection cell which would have resulted in at least the observed number of source counts. The value of  $p$  is calculated using the incomplete Gamma function  $P(a, x)$  as a function of raw source counts and raw background counts in the detection box. See Press et al., Numerical Recipes, chapter 6.2 for the calculation of  $P(a, x)$ .

$$LIKE = -\ln p(BOX\_CTS, BG\_RAW \cdot n^2)$$

In case of simultaneous detection runs over several energy bands, the LIKE values from each individual energy band are added and transformed to equivalent single band detection likelihoods using the incomplete Gamma function:



$$L = P(n, \sum_{i=1}^n L_i)$$

where  $n$  is the number of energy bands. A source is included in the output table, if the equivalent single band detection likelihood exceeds the threshold given by parameter `likemin`.

If detection over several energy bands is performed, up to three hardness ratios are calculated from the source counts in the individual bands. The hardness ratios are defined as follows:

$$HR_i = \frac{B_m - B_n}{B_m + B_n}$$

where  $B$  denotes the count rates in energy bands  $n$  and  $m$ , respectively.  $n$  and  $m$  are specified in input parameter `hrdef` for each of the three hardness ratios. The default band assignment is given in the following table:

$i$	$n$	$m$
1	1	2
2	2	3
3	3	4

The band numbers  $n$ , and  $m$  (output table column ID.BAND) are assigned to the individual bands by numbering the corresponding input images in the order in which they are given on the command line. It is therefore important that the ordering of the input images is consistent with the contents of `hrdef` to obtain meaningful hardness ratios.

## 4 Parameters

This section documents the parameters recognized by this task (if any).

Parameter	Mand	Type	Default	Constraints
-----------	------	------	---------	-------------

<b>imagesets</b>	yes	filename	image.fits	
------------------	-----	----------	------------	--

Names of EPIC fits images<sup>1,2</sup>

<b>boxlistset</b>	yes	filename	eboxlist.fits	
-------------------	-----	----------	---------------	--

Name of output source list

<b>bkgimagesets</b>	no	filename	bkgimage.fits	
---------------------	----	----------	---------------	--

Names of spline background maps<sup>1,2</sup>

<b>expimagesets</b>	no	filename	expimage.fits	
---------------------	----	----------	---------------	--

Names of exposure images<sup>1,2</sup>

<b>detmasksets</b>	no	filename	detmask.fits	
--------------------	----	----------	--------------	--

Name of detection mask<sup>3</sup>



<b>mergedlistset</b>	no	filename	mergedlist.fits	
----------------------	----	----------	-----------------	--

Name of merged source list

<b>usemap</b>	no	boolean	false	
---------------	----	---------	-------	--

true: map mode; false: local mode

<b>usematchedfilter</b>	no	boolean	false	
-------------------------	----	---------	-------	--

Matched filter flag; not supported in current version

<b>withdetmask</b>	no	boolean	false	
--------------------	----	---------	-------	--

Detection mask flag

<b>withexpimage</b>	no	boolean	false	
---------------------	----	---------	-------	--

Exposure image flag

<b>withoffsets</b>	no	boolean	false	
--------------------	----	---------	-------	--

Controls whether the merged source list will be read

<b>boxsize</b>	no	integer	5	3 or 5
----------------	----	---------	---	--------

Size of detection box (3 x 3 or 5 x 5 pixels)

<b>nruns</b>	no	integer	3	$1 \leq \text{param} \leq 4$
--------------	----	---------	---	------------------------------

Number of detection runs (detection box size is doubled after each run)

<b>likemin</b>	no	float	10.0	$[1.0 < \text{param} < 50.0]$
----------------	----	-------	------	-------------------------------

Minimum detection likelihood

<b>ecf</b>	no	float	1.0	$[0.001 < \text{param} < 1000]$
------------	----	-------	-----	---------------------------------

Energy conversion factors, given in units of  $10^{-11}$  counts  $\text{cm}^2 \text{erg}^{-1} \text{s}^{-1}$

<b>hrdef</b>	no	integer	1 2 2 3 3 4	0,10
--------------	----	---------	-------------	------

Array of six integer numbers specifying the upper and lower energy band for each of three hardness ratios

<b>pimin</b>	no	integer	2000	$[1 < \text{param} < 30000]$
--------------	----	---------	------	------------------------------

Lower PI channel limits of input images.<sup>2</sup>

<b>pimax</b>	no	integer	4500	$[1 < \text{param} < 30000]$
--------------	----	---------	------	------------------------------

Upper PI channel limits of input images.<sup>2</sup>

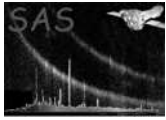
<sup>1</sup> Filenames must be ordered by instrument and energy band.

<sup>2</sup> One per energy band per instrument.

<sup>3</sup> One per instrument.

## 5 Errors

This section documents warnings and errors generated by this task (if any). Note that warnings and errors can also be generated in the SAS infrastructure libraries, in which case they would not be documented here. Refer to the index of all errors and warnings available in the HTML version of the SAS documentation.

**BufferFull** (*error*)

&gt; 10000 detections: source buffer overflow

**MissingParameter** (*error*)

Missing input file name

**FileMismatch** (*error*)

Inconsistent number of input images

**FileMismatch** (*error*)

Inconsistent instruments or bands

**WrongInst** (*error*)

Unknown instrument

**SizeMismatch** (*error*)

Image has wrong size

**WrongType** (*error*)

Input image has wrong type

**MissingAttribute** (*warning*)

Keyword is missing in input file

*corrective action:* Keyword is not copied to output file**Input image contains no photon (not yet implemented)** (*fatal*)**usemap true and background map contains values less or equal 0 within area marked for source searching (not yet implemented)** (*fatal*)**withexpimage true and exposure image contains values less or equal 0 within area marked for source searching (not yet implemented)** (*fatal*)**Input images, background maps, and exposure maps have non-matching telescope, instrument, and energy band keywords** (*fatal*)**Input images, background maps, detection mask, or exposure maps have different pixel sizes, image sizes, center coordinates (RA, Dec), or north vectors (not yet implemented)** (*fatal*)

## 6 Input Files

1. PPS product (from task **evselect**) : EPIC FITS images (one per instrument per energy band)
2. PPS product (from task **eexpmap**) : Exposure images (one per instrument)
3. from task **emask**: Detection mask
4. from task **esplinemap**: Spline background maps (one per instrument per energy band)



## 7 Output Files

1. PPS product (to be used by tasks **esplinemap** and **emldetect**) : EPIC **eboxdetect** source list

Description of source list columns:

Column name	Unit	Description
BOX_ID_SRC		Source index
ID_INST		Instrument (1:EPN, 2: MOS1, 3: MOS2)
ID_BAND		Index of input images per instrument, =0 for summary row
SCTS	cts	Net source counts corrected for PSF losses
SCTS_ERR	cts	Statistical error of SCTS
BOX_CTS	cts	Raw counts in detection box without background subtraction or PSF corrections.
X_IMA	image pix	X source position in image pixels
X_IMA_ERR	image pix	Statistical error of X_IMA
Y_IMA	image pix	Y source position in image pixels
Y_IMA_ERR	image pix	Statistical error of Y_IMA
LIKE*		Detection likelihood
BG_MAP	cts/pix	Background counts at source position
BG_RAW	cts/pix	Mean counts/pixel in background area, without correction for scattered source counts. For
EXP_MAP	seconds	Exposure at source position.
FLUX	erg/(cm <sup>2</sup> sec)	Flux calculated from RATE and parameter <b>ecf</b>
FLUX_ERR	erg/(cm <sup>2</sup> sec)	Statistical error of FLUX
RATE	cts/sec	Count rate (SCTS / EXP_MAP)
RATE_ERR	cts/sec	Error of RATE
RA	deg	Right ascension
DEC	deg	Declination
RADEC_ERR	deg	Statistical error of RA and DEC
LII	deg	Galactic longitude
BII	deg	Galactic latitude
HR1		Hardness ratio as defined via parameter <b>hrdef</b>
HR1_ERR		Statistical error of HR1
HR2		Hardness ratio as defined via parameter <b>hrdef</b>
HR2_ERR		Statistical error of HR2
HR3		Hardness ratio as defined via parameter <b>hrdef</b>
HR3_ERR		Statistical error of HR3
BOX_SIZE	pixel	Size of detect cell in image pixels
EEF		Enboxed energy fraction in detect cell. Calculated from the medium accuracy CAL PSF.
DIST_NN	arcsec	Distance to nearest neighbour.

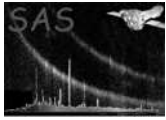
\* Note that column LIKE was named SIGMA in **eboxdetect** versions up to 4.8. The content of LIKE is identical to that of SIGMA in these older versions.

## 8 Algorithm

subroutine **eboxdetect**

Rebin images by linear factors of 1, 2, 4 ...  
(number of rebinning steps specified by parameter)

Loop over rebinned images:



```
Loop over image pixels:
  IF within area marked by detection mask
    (if mask flag is 'false': loop over all image pixels)
  THEN
    Loop over all instruments:
      Loop over all energy bands:

        IF local detect THEN
          Perform simple box search:
            1) Accumulate counts in 3 x 3 or 5 x 5 pixel source box.
            2) Accumulate counts in background area (16 background pixels
               from 5 x 5 box or 24 background pixels from 7 x 7 box
               centered on source box)
            3) Calculate statistical probability of excess in source box.
               Correct for border effects if source/background pixels
               are at image/detection mask border).
            4) Accumulate combined statistical probability of excesses
               in all energy bands of all instruments.
          END IF

          IF map detect THEN
            Perform matched filter/box search:
              1) Accumulate PSF weighted counts (PSF centred on 5 x 5 box)
                 and/or non-weighted counts (to handle extended objects)
                 in 5 x 5 pixel box.
              2) Calculate statistical probability of excess with respect
                 to background map. Correct for border effects if
                 source/background pixels are at image/detection mask
                 border.
              3) Accumulate combined statistical probability of excesses
                 in all energy bands of all instruments.
            END IF

          END IF
        END Loop
      END Loop

    IF combined statistical probability of excess exceeds threshold THEN
      Write source records to EBOXDETECT source table.
      Add fluxes and hardness ratios.
    END IF

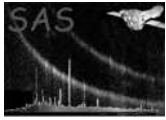
  END Loop
END Loop

end subroutine  eboxdetect
```

## 9 Comments

## 10 Future developments

- add matched filter option



## References